In [147…
```
import yfinance as yf
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error
import matplotlib.pyplot as plt
from datetime import timedelta
```

IMPORT DATA FROM YAHOO FINANCE

In [148…
```
# Define the ticker and date range
ticker = "^FTSE"  # FTSE 100 ticker symbol on Yahoo Finance
start_date = "2024-01-01"  # Start date
end_date = "2024-12-31"    # End date
```

In [149…
```
# Fetch data
ftse_data = yf.download(ticker, start=start_date, end=end_date)
```

```
[**********100%%**********]  1 of 1 completed
```

In [150…
```
# Display the first few rows
print(ftse_data.head())
```

```
                  Open         High          Low        Close    Adj Close  \
Date
2024-01-02  7733.200195  7764.399902  7689.000000  7721.500000  7721.500000
2024-01-03  7721.500000  7744.600098  7650.299805  7682.299805  7682.299805
2024-01-04  7682.299805  7728.200195  7678.799805  7723.100098  7723.100098
2024-01-05  7723.100098  7723.100098  7642.899902  7689.600098  7689.600098
2024-01-08  7689.600098  7694.200195  7636.100098  7694.200195  7694.200195

               Volume
Date
2024-01-02   678041300
2024-01-03   857615300
2024-01-04   963235800
2024-01-05   701524000
2024-01-08  1055586300
```

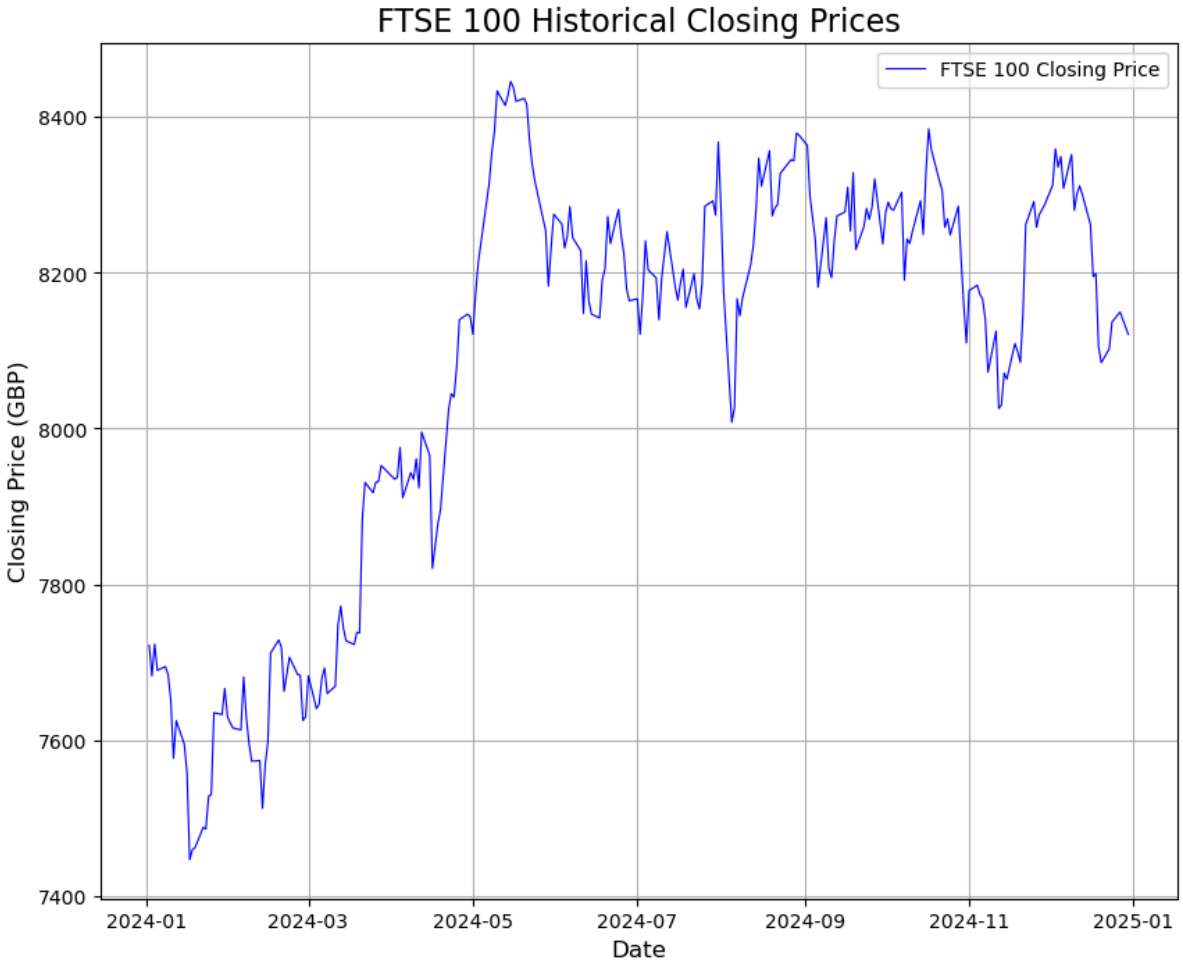
In [153…
```
print(ftse_data.describe()) # print summary statistics of FTSE 100
```

```
              Open         High          Low        Close    Adj Close  \
count   253.000000   253.000000   253.000000   253.000000   253.000000
mean   8075.352176  8112.967995  8042.843100  8077.101584  8077.101584
std     266.625331   266.448648   265.775858   265.846072   265.846072
min    7446.299805  7471.700195  7404.100098  7446.299805  7446.299805
25%    7911.200195  7938.100098  7882.700195  7917.600098  7917.600098
50%    8174.700195  8223.500000  8148.299805  8177.200195  8177.200195
75%    8276.700195  8306.200195  8245.700195  8276.700195  8276.700195
max    8445.799805  8474.400391  8427.900391  8445.799805  8445.799805

             Volume
count  2.530000e+02
mean   9.205321e+08
std    3.115854e+08
min    0.000000e+00
25%    7.359275e+08
50%    8.637630e+08
75%    1.071633e+09
max    2.813498e+09
```

PLOTTING GRAPH FOR CLOSING PRICE

In [154…

```
# Plot the Closing Prices
plt.figure(figsize=(10, 8))
plt.plot(ftse_data.index, ftse_data['Close'], label="FTSE 100 Closing Price", color

# Add Titles and Labels
plt.title("FTSE 100 Historical Closing Prices", fontsize=16)
plt.xlabel("Date", fontsize=12)
plt.ylabel("Closing Price (GBP)", fontsize=12)
plt.legend()
plt.grid(True)
plt.show()
```



Creating a new column for price difference

In [155…

```
ftse_data['PriceDiff'] = ftse_data['Close'].shift(-1) - ftse_data['Close']

print(ftse_data.head(20))
```

```
                  Open         High          Low        Close    Adj Close  \
Date
2024-01-02  7733.200195  7764.399902  7689.000000  7721.500000  7721.500000
2024-01-03  7721.500000  7744.600098  7650.299805  7682.299805  7682.299805
2024-01-04  7682.299805  7728.200195  7678.799805  7723.100098  7723.100098
2024-01-05  7723.100098  7723.100098  7642.899902  7689.600098  7689.600098
2024-01-08  7689.600098  7694.200195  7636.100098  7694.200195  7694.200195
2024-01-09  7694.200195  7717.500000  7675.100098  7684.000000  7684.000000
2024-01-10  7684.000000  7684.000000  7647.399902  7651.799805  7651.799805
2024-01-11  7651.799805  7693.899902  7576.600098  7576.600098  7576.600098
2024-01-12  7576.600098  7655.200195  7576.600098  7624.899902  7624.899902
2024-01-15  7624.899902  7637.799805  7578.299805  7594.899902  7594.899902
2024-01-16  7594.899902  7594.899902  7538.899902  7558.299805  7558.299805
2024-01-17  7558.299805  7558.299805  7404.100098  7446.299805  7446.299805
2024-01-18  7446.299805  7471.700195  7426.600098  7459.100098  7459.100098
2024-01-19  7459.100098  7521.100098  7450.899902  7461.899902  7461.899902
2024-01-22  7461.899902  7503.600098  7456.000000  7487.700195  7487.700195
2024-01-23  7487.700195  7525.100098  7465.100098  7485.700195  7485.700195
2024-01-24  7485.700195  7538.899902  7485.700195  7527.700195  7527.700195
2024-01-25  7527.700195  7544.500000  7507.000000  7529.700195  7529.700195
2024-01-26  7529.700195  7651.799805  7529.700195  7635.100098  7635.100098
2024-01-29  7635.100098  7664.899902  7630.799805  7632.700195  7632.700195

               Volume   PriceDiff
Date
2024-01-02   678041300  -39.200195
2024-01-03   857615300   40.800293
2024-01-04   963235800  -33.500000
2024-01-05   701524000    4.600098
2024-01-08  1055586300  -10.200195
2024-01-09   703141300  -32.200195
2024-01-10   668838800  -75.199707
2024-01-11  1306895000   48.299805
2024-01-12   794125500  -30.000000
2024-01-15   740769500  -36.600098
2024-01-16  1128049600 -112.000000
2024-01-17   838880100   12.800293
2024-01-18   883212400    2.799805
2024-01-19  1242202400   25.800293
2024-01-22   787927500   -2.000000
2024-01-23  1048392200   42.000000
2024-01-24   869218500    2.000000
2024-01-25   735927500  105.399902
2024-01-26  1046240900   -2.399902
2024-01-29   636218800   33.599609
```

Creating a new column for daily return

In [156…

```
ftse_data['DailyReturn'] = ftse_data['PriceDiff']/ftse_data['Close']

print(ftse_data.head(20))
```

Date	Open	High	Low	Close	Adj Close	Volume	PriceDiff	DailyReturn
2024-01-02	7733.200195	7764.399902	7689.000000	7721.500000	7721.500000	678041300	-39.200195	-0.005077
2024-01-03	7721.500000	7744.600098	7650.299805	7682.299805	7682.299805	857615300	40.800293	0.005311
2024-01-04	7682.299805	7728.200195	7678.799805	7723.100098	7723.100098	963235800	-33.500000	-0.004338
2024-01-05	7723.100098	7723.100098	7642.899902	7689.600098	7689.600098	701524000	4.600098	0.000598
2024-01-08	7689.600098	7694.200195	7636.100098	7694.200195	7694.200195	1055586300	-10.200195	-0.001326
2024-01-09	7694.200195	7717.500000	7675.100098	7684.000000	7684.000000	703141300	-32.200195	-0.004191
2024-01-10	7684.000000	7684.000000	7647.399902	7651.799805	7651.799805	668838800	-75.199707	-0.009828
2024-01-11	7651.799805	7693.899902	7576.600098	7576.600098	7576.600098	1306895000	48.299805	0.006375
2024-01-12	7576.600098	7655.200195	7576.600098	7624.899902	7624.899902	794125500	-30.000000	-0.003934
2024-01-15	7624.899902	7637.799805	7578.299805	7594.899902	7594.899902	740769500	-36.600098	-0.004819
2024-01-16	7594.899902	7594.899902	7538.899902	7558.299805	7558.299805	1128049600	-112.000000	-0.014818
2024-01-17	7558.299805	7558.299805	7404.100098	7446.299805	7446.299805	838880100	12.800293	0.001719
2024-01-18	7446.299805	7471.700195	7426.600098	7459.100098	7459.100098	883212400	2.799805	0.000375
2024-01-19	7459.100098	7521.100098	7450.899902	7461.899902	7461.899902	1242202400	25.800293	0.003458
2024-01-22	7461.899902	7503.600098	7456.000000	7487.700195	7487.700195	787927500	-2.000000	-0.000267
2024-01-23	7487.700195	7525.100098	7465.100098	7485.700195	7485.700195	1048392200	42.000000	0.005611
2024-01-24	7485.700195	7538.899902	7485.700195	7527.700195	7527.700195	869218500	2.000000	0.000266
2024-01-25	7527.700195	7544.500000	7507.000000	7529.700195	7529.700195	735927500	105.399902	0.013998
2024-01-26	7529.700195	7651.799805	7529.700195	7635.100098	7635.100098	1046240900	-2.399902	-0.000314
2024-01-29	7635.100098	7664.899902	7630.799805	7632.700195	7632.700195	636218800	33.599609	0.004402

Creating a new column for Direction using List Comprehension

In [157…

```
ftse_data['Direction'] = [1 if ftse_data['PriceDiff'].loc[ei] > 0 else 0 for ei in

print(ftse_data.head(20))
```

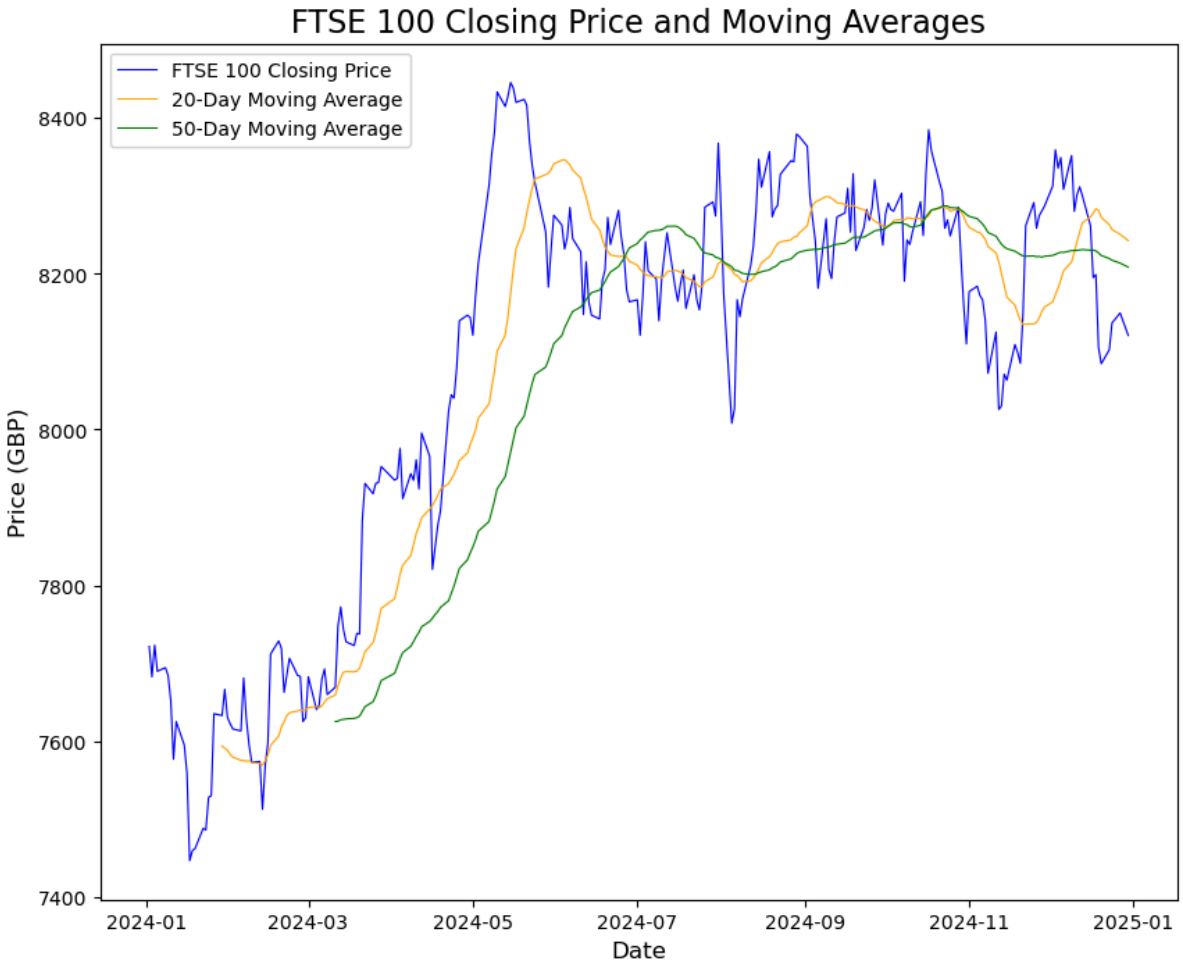
Date	Open	High	Low	Close	Adj Close	Volume	PriceDiff	DailyReturn	Direction
2024-01-02	7733.200195	7764.399902	7689.000000	7721.500000	7721.500000	678041300	-39.200195	-0.005077	0
2024-01-03	7721.500000	7744.600098	7650.299805	7682.299805	7682.299805	857615300	40.800293	0.005311	1
2024-01-04	7682.299805	7728.200195	7678.799805	7723.100098	7723.100098	963235800	-33.500000	-0.004338	0
2024-01-05	7723.100098	7723.100098	7642.899902	7689.600098	7689.600098	701524000	4.600098	0.000598	1
2024-01-08	7689.600098	7694.200195	7636.100098	7694.200195	7694.200195	1055586300	-10.200195	-0.001326	0
2024-01-09	7694.200195	7717.500000	7675.100098	7684.000000	7684.000000	703141300	-32.200195	-0.004191	0
2024-01-10	7684.000000	7684.000000	7647.399902	7651.799805	7651.799805	668838800	-75.199707	-0.009828	0
2024-01-11	7651.799805	7693.899902	7576.600098	7576.600098	7576.600098	1306895000	48.299805	0.006375	1
2024-01-12	7576.600098	7655.200195	7576.600098	7624.899902	7624.899902	794125500	-30.000000	-0.003934	0
2024-01-15	7624.899902	7637.799805	7578.299805	7594.899902	7594.899902	740769500	-36.600098	-0.004819	0
2024-01-16	7594.899902	7594.899902	7538.899902	7558.299805	7558.299805	1128049600	-112.000000	-0.014818	0
2024-01-17	7558.299805	7558.299805	7404.100098	7446.299805	7446.299805	838880100	12.800293	0.001719	1
2024-01-18	7446.299805	7471.700195	7426.600098	7459.100098	7459.100098	883212400	2.799805	0.000375	1
2024-01-19	7459.100098	7521.100098	7450.899902	7461.899902	7461.899902	1242202400	25.800293	0.003458	1
2024-01-22	7461.899902	7503.600098	7456.000000	7487.700195	7487.700195	787927500	-2.000000	-0.000267	0
2024-01-23	7487.700195	7525.100098	7465.100098	7485.700195	7485.700195	1048392200	42.000000	0.005611	1
2024-01-24	7485.700195	7538.899902	7485.700195	7527.700195	7527.700195	869218500	2.000000	0.000266	1
2024-01-25	7527.700195	7544.500000	7507.000000	7529.700195	7529.700195	735927500	105.399902	0.013998	1
2024-01-26	7529.700195	7651.799805	7529.700195	7635.100098	7635.100098	1046240900	-2.399902	-0.000314	0
2024-01-29	7635.100098	7664.899902	7630.799805	7632.700195	7632.700195	636218800	33.599609	0.004402	1

Create a new column in the DataFrame using Rolling Window calculation (.rolling()) - Moving average

In [159…

```
# Calculate Moving Averages
MA20 = 20  # Short-term moving average (e.g., 20 days)
MA50 = 50  # Long-term moving average (e.g., 50 days)

ftse_data['MA20'] = ftse_data['Close'].rolling(window= MA20).mean()
ftse_data['MA50'] = ftse_data['Close'].rolling(window= MA50).mean()

# Plot the Closing Price and Moving Averages
plt.figure(figsize=(10, 8))
plt.plot(ftse_data.index, ftse_data['Close'], label="FTSE 100 Closing Price", color
```

```
plt.plot(ftse_data.index, ftse_data['MA20'], label=f"{MA20}-Day Moving Average", co
plt.plot(ftse_data.index, ftse_data['MA50'], label=f"{MA50}-Day Moving Average", co

# Add Titles and Labels
plt.title("FTSE 100 Closing Price and Moving Averages", fontsize=16)
plt.xlabel("Date", fontsize=12)
plt.ylabel("Price (GBP)", fontsize=12)
plt.legend()

# Show the Plot
plt.show()
```



Building a Simple Trading Strategy

In [161…

```
# Calculate Moving Averages
ftse_data['MA20'] = ftse_data['Close'].rolling(window= MA20).mean()  # 20-day movir
ftse_data['MA50'] = ftse_data['Close'].rolling(window= MA50).mean()  # 50-day movir
ftse_data = ftse_data.dropna()

# Display the first few rows of the DataFrame
print(ftse_data.head(60))
```

Date	Open	High	Low	Close	Adj Close	Volume	PriceDiff	DailyReturn	Direction	MA20
2024-03-11	7659.700195	7669.200195	7612.600098	7669.200195	7669.200195					
2024-03-12	7669.200195	7764.600098	7669.200195	7747.799805	7747.799805					
2024-03-13	7747.799805	7785.700195	7738.700195	7772.200195	7772.200195					
2024-03-14	7772.200195	7778.399902	7719.100098	7743.200195	7743.200195					
2024-03-15	7743.200195	7761.100098	7727.399902	7727.399902	7727.399902					
2024-03-18	7727.399902	7750.200195	7714.899902	7722.600098	7722.600098					
2024-03-19	7722.600098	7739.700195	7699.600098	7738.299805	7738.299805					
2024-03-20	7738.299805	7749.399902	7711.500000	7737.399902	7737.399902					
2024-03-21	7737.399902	7901.799805	7737.399902	7882.600098	7882.600098					
2024-03-22	7882.600098	7961.399902	7882.200195	7930.899902	7930.899902					
2024-03-25	7930.899902	7939.399902	7891.700195	7917.600098	7917.600098					
2024-03-26	7917.600098	7934.399902	7892.100098	7931.000000	7931.000000					
2024-03-27	7931.000000	7938.100098	7893.000000	7932.000000	7932.000000					
2024-03-28	7932.000000	7975.399902	7931.399902	7952.600098	7952.600098					
2024-04-02	7952.600098	8015.600098	7928.600098	7935.100098	7935.100098					
2024-04-03	7935.100098	7937.399902	7882.700195	7937.399902	7937.399902					
2024-04-04	7937.399902	7990.399902	7937.399902	7975.899902	7975.899902					
2024-04-05	7975.899902	7975.899902	7884.500000	7911.200195	7911.200195					
2024-04-08	7911.200195	7953.200195	7887.899902	7943.500000	7943.500000					
2024-04-09	7943.500000	7962.799805	7917.500000	7934.799805	7934.799805					
2024-04-10	7934.799805	7999.799805	7916.500000	7961.200195	7961.200195					
2024-04-11	7961.200195	7970.100098	7886.899902	7923.799805	7923.799805					
2024-04-12	7923.799805	8045.000000	7923.799805	7995.600098	7995.600098					
2024-04-15	7995.600098	8009.500000	7952.100098	7965.500000	7965.500000					
2024-04-16	7965.500000	7965.500000	7793.899902	7820.399902	7820.399902					
2024-04-17	7820.399902	7893.600098	7798.600098	7848.000000	7848.000000					
2024-04-18	7848.000000	7898.799805	7845.899902	7877.100098	7877.100098					
2024-04-19	7877.100098	7900.500000	7809.700195	7895.899902	7895.899902					
2024-04-22	7895.899902	8042.799805	7895.899902	8023.899902	8023.899902					
2024-04-23	8023.899902	8076.500000	8021.700195	8044.799805	8044.799805					
2024-04-24	8044.799805	8092.200195	8031.799805	8040.399902	8040.399902					
2024-04-25	8040.399902	8105.600098	8030.100098	8078.899902	8078.899902					
2024-04-26	8078.899902	8146.799805	8078.899902	8139.799805	8139.799805					
2024-04-29	8139.799805	8189.100098	8136.700195	8147.000000	8147.000000					
2024-04-30	8147.000000	8200.000000	8138.100098	8144.100098	8144.100098					
2024-05-01	8144.100098	8179.000000	8111.399902	8121.200195	8121.200195					
2024-05-02	8121.200195	8178.799805	8119.899902	8172.200195	8172.200195					
2024-05-03	8172.200195	8248.700195	8172.200195	8213.500000	8213.500000					
2024-05-07	8213.500000	8335.700195	8213.500000	8313.700195	8313.700195					
2024-05-08	8313.700195	8365.299805	8313.700195	8354.099609	8354.099609					
2024-05-09	8354.099609	8396.299805	8349.799805	8381.400391	8381.400391					
2024-05-10	8381.400391	8455.799805	8381.400391	8433.799805	8433.799805					
2024-05-13	8433.799805	8446.500000	8409.200195	8415.000000	8415.000000					
2024-05-14	8415.000000	8448.700195	8407.000000	8428.099609	8428.099609					
2024-05-15	8428.099609	8474.400391	8427.900391	8445.799805	8445.799805					
2024-05-16	8445.799805	8455.700195	8406.700195	8438.700195	8438.700195					
2024-05-17	8438.700195	8441.700195	8401.400391	8420.299805	8420.299805					
2024-05-20	8420.299805	8451.599609	8418.000000	8424.200195	8424.200195					
2024-05-21	8424.200195	8424.200195	8379.099609	8416.500000	8416.500000					
2024-05-22	8416.500000	8416.500000	8343.400391	8370.299805	8370.299805					
2024-05-23	8370.299805	8385.200195	8332.299805	8339.200195	8339.200195					
2024-05-24	8339.200195	8339.200195	8263.799805	8317.599609	8317.599609					
2024-05-28	8317.599609	8335.500000	8241.299805	8254.200195	8254.200195					
2024-05-29	8254.200195	8255.400391	8180.000000	8183.100098	8183.100098					
2024-05-30	8183.100098	8236.299805	8148.500000	8231.099609	8231.099609					
2024-05-31	8231.099609	8289.200195	8231.099609	8275.400391	8275.400391					
2024-06-03	8275.400391	8371.700195	8258.900391	8262.799805	8262.799805					
2024-06-04	8262.799805	8263.400391	8200.299805	8232.000000	8232.000000					
2024-06-05	8232.000000	8276.599609	8232.000000	8247.000000	8247.000000					
2024-06-06	8247.000000	8287.099609	8247.000000	8285.299805	8285.299805					

Date						
2024-03-11	1203377500	78.599609	0.010249	1	7659.144995	
2024-03-12	1136566100	24.400391	0.003149	1	7670.919995	
2024-03-13	1336746300	-29.000000	-0.003731	0	7681.110010	
2024-03-14	973121100	-15.800293	-0.002041	0	7688.395020	
2024-03-15	2813498300	-4.799805	-0.000621	0	7689.180005	
2024-03-18	1034442400	15.699707	0.002033	1	7688.885010	
2024-03-19	1085005200	-0.899902	-0.000116	0	7689.839990	
2024-03-20	1032409800	145.200195	0.018766	1	7693.584985	
2024-03-21	1352537800	48.299805	0.006127	1	7703.489990	
2024-03-22	1056237300	-13.299805	-0.001677	0	7714.719995	
2024-03-25	1070739800	13.399902	0.001692	1	7726.385010	
2024-03-26	1278455500	1.000000	0.000126	1	7738.785010	
2024-03-27	1017867800	20.600098	0.002597	1	7754.135010	
2024-03-28	1051189700	-17.500000	-0.002201	0	7770.265015	
2024-04-02	1126449500	2.299805	0.000290	1	7782.895020	
2024-04-03	1131676800	38.500000	0.004850	1	7797.750024	
2024-04-04	1138495500	-64.699707	-0.008112	0	7814.235010	
2024-04-05	951509100	32.299805	0.004083	1	7825.830029	
2024-04-08	847327700	-8.700195	-0.001095	0	7838.380029	
2024-04-09	1000931200	26.400391	0.003327	1	7852.135010	
2024-04-10	1207706000	-37.400391	-0.004698	0	7866.735010	
2024-04-11	1197883800	71.800293	0.009061	1	7875.535010	
2024-04-12	937766700	-30.100098	-0.003765	0	7886.705005	
2024-04-15	820441500	-145.100098	-0.018216	0	7897.819995	
2024-04-16	1058250300	27.600098	0.003529	1	7902.469995	
2024-04-17	939573900	29.100098	0.003708	1	7908.739990	
2024-04-18	1222486300	18.799805	0.002387	1	7915.680005	
2024-04-19	982567400	128.000000	0.016211	1	7923.605005	
2024-04-22	1085332800	20.899902	0.002605	1	7930.669995	
2024-04-23	1082600700	-4.399902	-0.000547	0	7936.364990	
2024-04-24	1420290700	38.500000	0.004788	1	7942.504980	
2024-04-25	1224259900	60.899902	0.007538	1	7949.899976	
2024-04-26	904872200	7.200195	0.000885	1	7960.289966	
2024-04-29	759963000	-2.899902	-0.000356	0	7970.009961	
2024-04-30	868339500	-22.899902	-0.002812	0	7980.459961	
2024-05-01	562235500	51.000000	0.006280	1	7989.649976	
2024-05-02	919730600	41.299805	0.005054	1	7999.464990	
2024-05-03	658456400	100.200195	0.012199	1	8014.579980	
2024-05-07	1220562500	40.399414	0.004859	1	8033.089990	
2024-05-08	1272316300	27.300781	0.003268	1	8054.054980	
2024-05-09	1079046000	52.399414	0.006252	1	8075.064990	
2024-05-10	698222200	-18.799805	-0.002229	0	8100.564990	
2024-05-13	974791500	13.099609	0.001557	1	8121.534985	
2024-05-14	1160778700	17.700195	0.002100	1	8144.664966	
2024-05-15	1112620600	-7.099609	-0.000841	0	8175.934961	
2024-05-16	889766900	-18.400391	-0.002180	0	8205.469971	
2024-05-17	1034395100	3.900391	0.000463	1	8232.629956	
2024-05-20	739021000	-7.700195	-0.000914	0	8259.044971	
2024-05-21	1001396500	-46.200195	-0.005489	0	8278.674976	
2024-05-22	1286558300	-31.099609	-0.003715	0	8294.949976	
2024-05-23	1229161600	-21.600586	-0.002590	0	8309.889990	
2024-05-24	779070900	-63.399414	-0.007622	0	8321.824976	
2024-05-28	1202410100	-71.100098	-0.008614	0	8327.544995	
2024-05-29	1014436000	47.999512	0.005866	1	8329.350000	
2024-05-30	985401500	44.300781	0.005382	1	8333.699976	
2024-05-31	2232006600	-12.600586	-0.001523	0	8341.409985	
2024-06-03	1095687900	-30.799805	-0.003728	0	8345.939966	
2024-06-04	1127695300	15.000000	0.001822	1	8346.864966	
2024-06-05	879057500	38.299805	0.004644	1	8343.529956	
2024-06-06	857611200	-39.899414	-0.004816	0	8340.089966	

	MA50
Date	

2024-03-11 7624.982012
2024-03-12 7625.508008
2024-03-13 7627.306016
2024-03-14 7627.708018
2024-03-15 7628.464014
2024-03-18 7629.032012
2024-03-19 7630.118008
2024-03-20 7631.830010
2024-03-21 7637.950010
2024-03-22 7644.070010
2024-03-25 7650.524014
2024-03-26 7657.978018
2024-03-27 7667.692021
2024-03-28 7677.562021
2024-04-02 7687.026025
2024-04-03 7696.020020
2024-04-04 7705.824014
2024-04-05 7713.494014
2024-04-08 7721.770010
2024-04-09 7727.764004
2024-04-10 7734.334004
2024-04-11 7739.484004
2024-04-12 7746.784004
2024-04-15 7753.650000
2024-04-16 7757.747998
2024-04-17 7762.450000
2024-04-18 7766.372002
2024-04-19 7771.714004
2024-04-22 7780.282002
2024-04-23 7789.725996
2024-04-24 7799.059990
2024-04-25 7810.391992
2024-04-26 7821.819990
2024-04-29 7832.809990
2024-04-30 7841.457988
2024-05-01 7849.311992
2024-05-02 7858.371992
2024-05-03 7869.391992
2024-05-07 7881.975996
2024-05-08 7894.931992
2024-05-09 7908.874004
2024-05-10 7923.890000
2024-05-13 7939.690000
2024-05-14 7955.651992
2024-05-15 7970.917988
2024-05-16 7986.885996
2024-05-17 8002.367988
2024-05-20 8017.265996
2024-05-21 8031.745996
2024-05-22 8045.957988
2024-05-23 8059.357988
2024-05-24 8070.753984
2024-05-28 8080.393984
2024-05-29 8089.191982
2024-05-30 8099.265977
2024-05-31 8110.321982
2024-06-03 8120.811982
2024-06-04 8130.703984
2024-06-05 8137.991982
2024-06-06 8145.079980

Add a new column "Shares", if MA20 > MA50, denote as 1 (long one share of stock), otherwise, denote as 0 (do nothing)

In [162…

```
# Add Shares Column: 1 if MA10 > MA50, else 0
ftse_data['Shares'] = (ftse_data['MA20'] > ftse_data['MA50']).astype(int)

# Display the first few rows of the DataFrame
print(ftse_data[['Close', 'MA20', 'MA50', 'Shares']].head(60))  # Showing 60 rows t
```

Date	Close	MA20	MA50	Shares
2024-03-11	7669.200195	7659.144995	7624.982012	1
2024-03-12	7747.799805	7670.919995	7625.508008	1
2024-03-13	7772.200195	7681.110010	7627.306016	1
2024-03-14	7743.200195	7688.395020	7627.708018	1
2024-03-15	7727.399902	7689.180005	7628.464014	1
2024-03-18	7722.600098	7688.885010	7629.032012	1
2024-03-19	7738.299805	7689.839990	7630.118008	1
2024-03-20	7737.399902	7693.584985	7631.830010	1
2024-03-21	7882.600098	7703.489990	7637.950010	1
2024-03-22	7930.899902	7714.719995	7644.070010	1
2024-03-25	7917.600098	7726.385010	7650.524014	1
2024-03-26	7931.000000	7738.785010	7657.978018	1
2024-03-27	7932.000000	7754.135010	7667.692021	1
2024-03-28	7952.600098	7770.265015	7677.562021	1
2024-04-02	7935.100098	7782.895020	7687.026025	1
2024-04-03	7937.399902	7797.750024	7696.020020	1
2024-04-04	7975.899902	7814.235010	7705.824014	1
2024-04-05	7911.200195	7825.830029	7713.494014	1
2024-04-08	7943.500000	7838.380029	7721.770010	1
2024-04-09	7934.799805	7852.135010	7727.764004	1
2024-04-10	7961.200195	7866.735010	7734.334004	1
2024-04-11	7923.799805	7875.535010	7739.484004	1
2024-04-12	7995.600098	7886.705005	7746.784004	1
2024-04-15	7965.500000	7897.819995	7753.650000	1
2024-04-16	7820.399902	7902.469995	7757.747998	1
2024-04-17	7848.000000	7908.739990	7762.450000	1
2024-04-18	7877.100098	7915.680005	7766.372002	1
2024-04-19	7895.899902	7923.605005	7771.714004	1
2024-04-22	8023.899902	7930.669995	7780.282002	1
2024-04-23	8044.799805	7936.364990	7789.725996	1
2024-04-24	8040.399902	7942.504980	7799.059990	1
2024-04-25	8078.899902	7949.899976	7810.391992	1
2024-04-26	8139.799805	7960.289966	7821.819990	1
2024-04-29	8147.000000	7970.009961	7832.809990	1
2024-04-30	8144.100098	7980.459961	7841.457988	1
2024-05-01	8121.200195	7989.649976	7849.311992	1
2024-05-02	8172.200195	7999.464990	7858.371992	1
2024-05-03	8213.500000	8014.579980	7869.391992	1
2024-05-07	8313.700195	8033.089990	7881.975996	1
2024-05-08	8354.099609	8054.054980	7894.931992	1
2024-05-09	8381.400391	8075.064990	7908.874004	1
2024-05-10	8433.799805	8100.564990	7923.890000	1
2024-05-13	8415.000000	8121.534985	7939.690000	1
2024-05-14	8428.099609	8144.664966	7955.651992	1
2024-05-15	8445.799805	8175.934961	7970.917988	1
2024-05-16	8438.700195	8205.469971	7986.885996	1
2024-05-17	8420.299805	8232.629956	8002.367988	1
2024-05-20	8424.200195	8259.044971	8017.265996	1
2024-05-21	8416.500000	8278.674976	8031.745996	1
2024-05-22	8370.299805	8294.949976	8045.957988	1
2024-05-23	8339.200195	8309.889990	8059.357988	1
2024-05-24	8317.599609	8321.824976	8070.753984	1
2024-05-28	8254.200195	8327.544995	8080.393984	1
2024-05-29	8183.100098	8329.350000	8089.191982	1
2024-05-30	8231.099609	8333.699976	8099.265977	1
2024-05-31	8275.400391	8341.409985	8110.321982	1
2024-06-03	8262.799805	8345.939966	8120.811982	1
2024-06-04	8232.000000	8346.864966	8130.703984	1
2024-06-05	8247.000000	8343.529956	8137.991982	1
2024-06-06	8285.299805	8340.089966	8145.079980	1

```
C:\Users\Ashlesh Sonde\AppData\Local\Temp\ipykernel_23708\3241852225.py:2: Setting
WithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stabl
e/user_guide/indexing.html#returning-a-view-versus-a-copy
  ftse_data['Shares'] = (ftse_data['MA20'] > ftse_data['MA50']).astype(int)
```

Adding a new column "Profit" using List Comprehension, for any rows in FTSE, if Shares = 1, the profit is calculated as " the close price of tomorrow - the close price of today". Otherwise the profit is 0.

In [164…

```python
# Calculate Profit Column Using List Comprehension
ftse_data['Close1'] = ftse_data['Close'].shift(-1)
ftse_data['Profit'] = [ftse_data.loc[ei, 'Close1'] - ftse_data.loc[ei, 'Close'] if

# Display the first few rows
print(ftse_data[['Close', 'MA20', 'MA50', 'Shares', 'Profit']].head(60))  # Show mo
```

Date	Close	MA20	MA50	Shares	Profit
2024-03-11	7669.200195	7659.144995	7624.982012	1	78.599609
2024-03-12	7747.799805	7670.919995	7625.508008	1	24.400391
2024-03-13	7772.200195	7681.110010	7627.306016	1	-29.000000
2024-03-14	7743.200195	7688.395020	7627.708018	1	-15.800293
2024-03-15	7727.399902	7689.180005	7628.464014	1	-4.799805
2024-03-18	7722.600098	7688.885010	7629.032012	1	15.699707
2024-03-19	7738.299805	7689.839990	7630.118008	1	-0.899902
2024-03-20	7737.399902	7693.584985	7631.830010	1	145.200195
2024-03-21	7882.600098	7703.489990	7637.950010	1	48.299805
2024-03-22	7930.899902	7714.719995	7644.070010	1	-13.299805
2024-03-25	7917.600098	7726.385010	7650.524014	1	13.399902
2024-03-26	7931.000000	7738.785010	7657.978018	1	1.000000
2024-03-27	7932.000000	7754.135010	7667.692021	1	20.600098
2024-03-28	7952.600098	7770.265015	7677.562021	1	-17.500000
2024-04-02	7935.100098	7782.895020	7687.026025	1	2.299805
2024-04-03	7937.399902	7797.750024	7696.020020	1	38.500000
2024-04-04	7975.899902	7814.235010	7705.824014	1	-64.699707
2024-04-05	7911.200195	7825.830029	7713.494014	1	32.299805
2024-04-08	7943.500000	7838.380029	7721.770010	1	-8.700195
2024-04-09	7934.799805	7852.135010	7727.764004	1	26.400391
2024-04-10	7961.200195	7866.735010	7734.334004	1	-37.400391
2024-04-11	7923.799805	7875.535010	7739.484004	1	71.800293
2024-04-12	7995.600098	7886.705005	7746.784004	1	-30.100098
2024-04-15	7965.500000	7897.819995	7753.650000	1	-145.100098
2024-04-16	7820.399902	7902.469995	7757.747998	1	27.600098
2024-04-17	7848.000000	7908.739990	7762.450000	1	29.100098
2024-04-18	7877.100098	7915.680005	7766.372002	1	18.799805
2024-04-19	7895.899902	7923.605005	7771.714004	1	128.000000
2024-04-22	8023.899902	7930.669995	7780.282002	1	20.899902
2024-04-23	8044.799805	7936.364990	7789.725996	1	-4.399902
2024-04-24	8040.399902	7942.504980	7799.059990	1	38.500000
2024-04-25	8078.899902	7949.899976	7810.391992	1	60.899902
2024-04-26	8139.799805	7960.289966	7821.819990	1	7.200195
2024-04-29	8147.000000	7970.009961	7832.809990	1	-2.899902
2024-04-30	8144.100098	7980.459961	7841.457988	1	-22.899902
2024-05-01	8121.200195	7989.649976	7849.311992	1	51.000000
2024-05-02	8172.200195	7999.464990	7858.371992	1	41.299805
2024-05-03	8213.500000	8014.579980	7869.391992	1	100.200195
2024-05-07	8313.700195	8033.089990	7881.975996	1	40.399414
2024-05-08	8354.099609	8054.054980	7894.931992	1	27.300781
2024-05-09	8381.400391	8075.064990	7908.874004	1	52.399414
2024-05-10	8433.799805	8100.564990	7923.890000	1	-18.799805
2024-05-13	8415.000000	8121.534985	7939.690000	1	13.099609
2024-05-14	8428.099609	8144.664966	7955.651992	1	17.700195
2024-05-15	8445.799805	8175.934961	7970.917988	1	-7.099609
2024-05-16	8438.700195	8205.469971	7986.885996	1	-18.400391
2024-05-17	8420.299805	8232.629956	8002.367988	1	3.900391
2024-05-20	8424.200195	8259.044971	8017.265996	1	-7.700195
2024-05-21	8416.500000	8278.674976	8031.745996	1	-46.200195
2024-05-22	8370.299805	8294.949976	8045.957988	1	-31.099609
2024-05-23	8339.200195	8309.889990	8059.357988	1	-21.600586
2024-05-24	8317.599609	8321.824976	8070.753984	1	-63.399414
2024-05-28	8254.200195	8327.544995	8080.393984	1	-71.100098
2024-05-29	8183.100098	8329.350000	8089.191982	1	47.999512
2024-05-30	8231.099609	8333.699976	8099.265977	1	44.300781
2024-05-31	8275.400391	8341.409985	8110.321982	1	-12.600586
2024-06-03	8262.799805	8345.939966	8120.811982	1	-30.799805
2024-06-04	8232.000000	8346.864966	8130.703984	1	15.000000
2024-06-05	8247.000000	8343.529956	8137.991982	1	38.299805
2024-06-06	8285.299805	8340.089966	8145.079980	1	-39.899414

```
C:\Users\Ashlesh Sonde\AppData\Local\Temp\ipykernel_23708\1440343168.py:2: Setting
WithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stabl
e/user_guide/indexing.html#returning-a-view-versus-a-copy
  ftse_data['Close1'] = ftse_data['Close'].shift(-1)
C:\Users\Ashlesh Sonde\AppData\Local\Temp\ipykernel_23708\1440343168.py:3: Setting
WithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stabl
e/user_guide/indexing.html#returning-a-view-versus-a-copy
  ftse_data['Profit'] = [ftse_data.loc[ei, 'Close1'] - ftse_data.loc[ei, 'Close']
if ftse_data.loc[ei, 'Shares']==1 else 0 for ei in ftse_data.index]
```

Calculating Cumulative Profit (Wealth Accmulated)

In [168…

```
ftse_data['wealth'] = ftse_data['Profit'].cumsum()
ftse_data.tail(10)
```

```
C:\Users\Ashlesh Sonde\AppData\Local\Temp\ipykernel_23708\935832223.py:2: SettingW
ithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stabl
e/user_guide/indexing.html#returning-a-view-versus-a-copy
  ftse_data['wealth'] = ftse_data['Profit'].cumsum()
```

Out[168]:

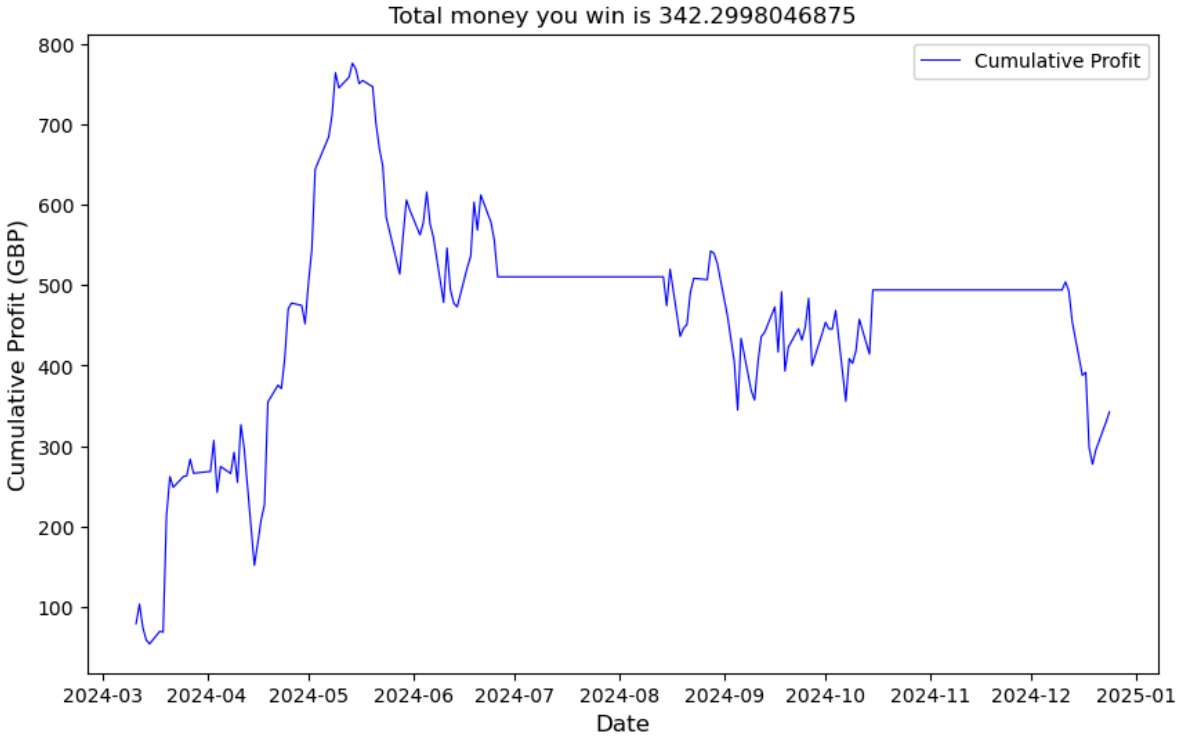
Date	Open	High	Low	Close	Adj Close	Volume	PriceDiff	I
2024-12-12	8301.599609	8331.099609	8300.200195	8311.799805	8311.799805	584951900	-11.500000	
2024-12-13	8311.799805	8331.900391	8282.400391	8300.299805	8300.299805	653230000	-38.200195	
2024-12-16	8300.299805	8305.200195	8254.099609	8262.099609	8262.099609	621726000	-66.899414	
2024-12-17	8262.099609	8262.099609	8190.200195	8195.200195	8195.200195	1176055700	3.899414	
2024-12-18	8195.200195	8231.799805	8191.600098	8199.099609	8199.099609	836400800	-93.799805	
2024-12-19	8199.099609	8199.099609	8079.000000	8105.299805	8105.299805	802810600	-20.699707	
2024-12-20	8105.299805	8105.799805	8002.299805	8084.600098	8084.600098	1608807700	18.100098	
2024-12-23	8084.600098	8111.399902	8051.899902	8102.700195	8102.700195	573891100	34.299805	
2024-12-24	8102.700195	8152.000000	8102.700195	8137.000000	8137.000000	243246600	12.799805	
2024-12-27	8137.000000	8151.299805	8112.299805	8149.799805	8149.799805	356239100	-28.799805	

In [173…

```
# Plot the Cumulative Profit Over Time
plt.figure(figsize=(10, 6))
plt.plot(ftse_data.index, ftse_data['wealth'], label="Cumulative Profit", color='bl

# Add Titles and Labels
plt.title('Total money you win is {}'.format(ftse_data.loc[ftse_data.index[-2], 'we
plt.xlabel("Date", fontsize=12)
plt.ylabel("Cumulative Profit (GBP)", fontsize=12)
plt.legend()
```

Out[173]:

```
<matplotlib.legend.Legend at 0x25659dfbbd0>
```


Total money you win is 342.2998046875
Cumulative Profit
Cumulative Profit (GBP)
800
700
600
500
400
300
200
100
2024-03
2024-04
2024-05
2024-06
2024-07
2024-08
2024-09
2024-10
2024-11
2024-12
2025-01
Date